SNAPSHOT 64

Instruction Manual

CSM SOFTWARE, INC.

TABLE OF CONTENTS

NOTE

We realize that some of you won't be able to wait long enough to read this manual thoroughly before trying out your newest acquisition. So for those of you who fall into this category, GO DIRECTLY TO PAGE 7 and read the section called 'USING SNAPSHOT64'.

USING SNAPSHOT64

SNAPSHOT64 INTRODUCTION

Congratulations and thank you for your purchase of SNAPSHOT64. We hope that you find it useful and are confident that you will be very satisfied with it.

To make full use of your SNAPSHOT64 system (and to save you needless frustration), please read the documentation carefully. A few minutes should be enough to totally familiarize yourself with the correct procedures to operate the system. Many hours have been spent to make SNAPSHOT64 as simple and user friendly as possible. So to help yourself (and make us believe that we didn't waste money on this paper) please spend a few moments and read this manual.

HOW IT WORKS

The premise of SNAPSHOT64 is unique and really very simple (the premise, not the programming). When a program loads, the protection scheme is USUALLY (see CAPABILITIES) executed during the load. If the protection is found then the program continues to load. If it is not, well, we all know what happens then. The program that is put into memory is usually free of the protection scheme. This is the point where SNAPSHOT64 comes in. Although SNAPSHOT64 is a cartridge that plugs into the 64, until it is activated it is completely invisible.

Not even the computer knows that it is there! With the simple push of a button and the press of a key, SNAPSHOT64 is activated. It immediately seizes control of the computer and begins to analyze the code contained therein.

Its first function is to locate all of the relevant code necessary to the proper functioning of the program. When this is accomplished it downloads the selected memory into a small number of convenient files complete with an auto-loading boot. These files can then be loaded from any 64 compatible drive.

The broken program will load faster than the original especially if used with one of the commercially available 'fast loaders'. The program can also be saved on a disk with other programs thus allowing you to make more efficient use of your available disk space.

REQUIREMENTS

SNAPSHOT64 requires a Commodore 64 and a compatible disk drive. Usually a 1541 works best as many copy-protected programs will not load on anything but a 1541. SNAPSHOT64 also works with an SX-64. The package that you received should contain the following components:

SNAPSHOT64 cartridge
Documentation
Template
Warranty registration card

SET UP PROCEDURE

Here is where you will really begin to appreciate the ease of use of this product. Simply TURN OFF your 64 and then insert the SNAPSHOT64 cartridge. Now turn on your computer and then the disk drive. That's all there is to it! There are no special programs to load or switches to check. You're ready for your first encounter.

DESCRIPTION OF TERMS USED

The following is a list of some of the terms used that initially may be found to be confusing:

1) SNAPSHOT64...the cartridge that you paid that ridiculously small amount for.

2) Original disk...the original program disk.

3) Target disk...the disk that will contain the broken version of the program.

4) Template...the cardboard overlay that came with your package. This is designed to fit over your function keys and to remind you of the function of the various keys.

DESCRIPTION OF FUNCTION KEYS

Before you go racing off to tackle that first conquest you should familiarize yourself with the purpose of the function keys and the button on the cartridge. The following is a brief overview:

Cartridge button--This 'freezes' the computer at the screen currently displayed.

F1--Pressing this key instructs SNAPSHOT64 to begin the process of analyzing and saving the program.

F3--This key clears the computer's memory which is necessary for the minimizing function.

F5--Fondly referred to as the panic button. It is used when you have begun the SNAPSHOT64 process by loading in the original program and realize that, much to your horror, you do not have a blank formatted disk and are not sure that you have a disk lying around that has at least 273 blocks free (the most space that the total SNAPSHOT64 files could take). Pressing this key will format a disk with a pre-determined header and a random ID. Press this button instead of F1.

F7--This is the abort/resume key. It is used when you mistakenly push the cartridge button before the desired screen has appeared. Most of the time the program will continue after this key

is pressed. However, sometimes this will not work due to the way that the original was written and you will have to start all over again.

CAPABILITIES

SNAPSHOT64 will work on virtually all software programs that load entirely into the computer's memory. There are some programs that cannot be defeated by it, however. Problems will be encountered with the following types:

1) Programs that do a protection check well into the game.

2) Programs that store code in the 1541's memory.

3) Programs that are dongle, cartridge or key protected.

4) Programs that refer back to the disk and expect to find the information at a specific track and sector. There IS a cure for this. Simply do a straight backup of the original and place it on another disk or on the flip side of the disk that has the files produced by SNAPSHOT64.

If the program simply refers back to the disk for file information and does not care where it is found, all you have to do is copy those files onto the target disk after it has the files produced by SNAPSHOT64. To do this, use any of the

many file copy utilities that are available commercially or public domain. You could use the file copier that came on your demo disk with your disk drive.

It should be pointed out again that it is a very small percentage of programs that cannot be successfully copied using SNAPSHOT64.

If you wish to find out if a particular program contains a type of protection that SNAPSHOT64 cannot get past, try using the following procedures.

Protection that is checked after the program has loaded and begun can be detected in the following manner. After the program has loaded and the drive is INACTIVE, replace the original disk with a straight backup of the program. DO NOT attempt to duplicate the protection of the original. Continue using the program and watch for the first time that the program accesses the drive. If the program hangs up or does not act correctly then you can assume that a protection scheme has been checked for and not found. On the other hand, if everything appears to work normally, then you know that there is no protection scheme left.

Protection that resides in the RAM of the disk drive can be determined with equal ease. First load the original program as you would normally. When the program stops, at an option screen for instance, try turning off your drive and then turning it on again. Choose an option and watch to see if the program

can successfully access the drive. If something strange happens, such as the computer hanging up or the screen displaying garbage, you can safely assume that the program is using drive protection. If the program accesses the drive successfully there should be no problem copying the program using SNAPSHOT64.

USING SNAPSHOT64

The following steps should be used when copying a program with SNAPSHOT64:

1) Place the Template over the function keys.

2) Make sure you have a blank formatted disk or a disk with at least 273 blocks free. This is your target disk.

3) With the cartridge inserted (see SETUP PROCEDURE) press the button on the SNAPSHOT64 cartridge. You will notice that the screen goes blank.

4) Press F3. The screen should come back with the message 'CLEARED' and a blinking cursor.

5) Load in the original disk as you would normally. *NOTE* Please ensure that your original program disk is write protected as you may inadvertently insert the wrong disk or simply forget to change disks when prompted. Mistakes do happen.

6) Press the button on the SNAPSHOT64 cartridge when the program has loaded and the first option screen appears (preferably one with little music or movement although this is not crucial and is for simple asthetics). The screen will blank to the current border color. At this time remove the original disk and turn your drive off and then on again. If your using an SX push the drive reset.

7) Insert your target disk and then press the F1 key. The drive will come on and after approximately 25 seconds the screen will turn dark blue and the message 'WORKING...' will appear. If you have forgotten to format a new disk press F5 instead of F1. In this case the drive will activate and format a disk first. Then the blue screen and message will appear.

8) Within two to three minutes (depending upon the size of the program) a message will appear on the screen prompting you for two characters by which to identify the new files. Type in the two characters and then press return. The drive will again activate. The message '00,ok,00,00' should now appear. This simply tells you that the drive channel is clear and drive functions will be successful. Now the message 'RENAMING FILES...' will appear on the screen.

9) In approximately twenty seconds another message will appear asking for a name for the new boot. This name must be sixteen characters or less. We would suggest that you use the original name of the program. Once you have typed in the

desired name press return. The drive will come on again and the drive channel status will be displayed. Shortly thereafter the drive will stop and the message 'SNAPSHOT64 PROCESS COMPLETED' appear.

YOU'RE DONE!!

SNAPSHOT64 has now finished and again is invisible. At this point you can load in the new program by typing-- LOAD "name of program",8,1-- and pressing return (if there are no extra files to be copied by way of a separate file copier).

EXTRA FEATURE

There is special program built into SNAPSHOT64 that we call CODE INSPECTOR. This will only be of benefit to those familiar with machine language (hereafter called ML). It is meant to be a tool that can be used to tackle those programs that cannot be broken by SNAPSHOT64.

This feature is accessed by first pressing the button on the cartridge. The screen will freeze and default to the border color. Pressing the RETURN key will turn the screen black and the current status of the following will be displayed:

1) Program Counter   (PC)
2) IRQ Vector
3) NMI Vector
4) Processor Status Register (SR)

5) Accumulator   (AC)
6) X Register   (XR)
7) Y Register   (YR)
8) Stack Pointer   (SP)

Also displayed will be the contents of the following:

9) 6510 on-chip I/O registers (0&1)
10) Vic II Chip IRQ Mask & Raster
    Compare latch value
11) CIA ICR mask values and timer
    latch values

There will be four options displayed underneath this information:

The first (L) will allow you to load and execute a machine language file from disk. This could be used to load in a ML monitor or perhaps a custom program. It will ask for the name of the file and then for the jump address. So, if for example, you wanted to load a monitor called MYMON and the entry point was 49152 you would press L. When the first prompt appeared you would type MYMON and press return. On the second prompt you would type C000 (hex for 49152).

The second option (J) will be to jump to any address point in memory. This could be used to test for proper entry points for program start up. You would type J and then enter the desired address when prompted.

Option number three (D) will allow you to display memory in hex. Enter the start and end addresses when prompted.

The dump will be to screen only.

The fourth option (B) will return you to basic.

As with SNAPSHOT64's main feature, the memory snapshot option, CODE INSPECTOR may require that you reset your disk drive before it will operate properly. The purpose of the CODE INSPECTOR is to allow you to easily and quickly examine the conditions inside your computer at the time the cartridge button was pressed. If all you want to do is see the status of your computer without modifying the contents of memory, you can make a quick exit using the RESUME option (hit J for çJ¿ump, then R and RETURN to resume). Keep in mind that some RAM had to be destroyed to make room for CODE INSPECTOR (approx. $0808 to $0CFF), and that the text screen at $0400-$07FF was rewritten. All these areas are saved uncorrupted when a memory snapshot is dumped to disk (in the normal SNAPSHOT64 process).

If you wish to make changes to the contents of memory once CODE INSPECTOR has been activated (to get around a late protection check, for example), be careful to locate your monitor in an area of memory that will not corrupt the SNAPSHOT64 file that will contain the desired changes. To simplify this task, we suggest that you take the time to identify the SNAPSHOT64 file that will be changed by your modifications in the table below, and make sure that its entire span of memory will remain uncorrupted by your monitor or other

SNAPSHOT64 FILE LOCATION

Load Address	Filename	Original Address
$0400	XXa	$0400-$07FF
$D800	XXb	$D800-$DBFF
$0800	XXc	$0800-$0FFF
$0F10	XXd	$1000-end of file; now in compressed format; was $1000-$87FF
$0F10	XXe	$1000-end of file; now in compressed format; was $8800-$FFFF
$0820	XXf	none-main boot
$0BB3	XXg	SID: $0BB3-$0BCF; moved to $D400
		VIC: $0BD0-$0BFE; moved to $D000
		SP: stack pointer; was $0C00-$0FFF; moved to $0000-$03FF

You can either make changes directly to the SNAPSHOT64 snapshot files, or make changes in memory via CODE INSPECTOR and an appropriate monitor.

When using the CODE INSPECTOR/monitor approach, make a SNAPSHOT64 copy of the program on one disk. Then reload the ORIGINAL disk and interrupt the program at the desired time with the CODE INSPECTOR. Make the desired changes to memory with a monitor and then use SNAPSHOT64 to make a new snapshot on a second disk. Scratch the appropriate file on disk one and copy the modified file from the second disk onto the first disk.

Used properly, CODE INSPECTOR can be a potent tool. Whether used for examining the programs of others or to debug your own programs, CODE INSPECTOR soon becomes an indespensible tool to ML programmers.

If you discover new ways of using SNAPSHOT64/CODE INSPECTOR, please feel free to get in touch with us. Your comments, suggestions or solutions are always appreciated. We'll try to spread the word around!

COPYRIGHT NOTICE

SNAPSHOT64 concepts by Marcel LeBlanc and Ron Smith

Programming by Marcel Leblanc
SNAPSHOT64 board layout by Cadmi
Microelectronics Ltd., Fredericton, N.B.
Manual by Ron Smith
Software,hardware and manual Copyright
(c) 1985 LMS TECHNOLOGIES

Thanks to Marshview Software for the
support and help.

And you too, Rob!

LIMITED WARRANTY

Neither the authors nor the
distributors of this product shall be
liable for any damages which may be
caused by any errors or omissions in this
product. Should the product be
defective, the distributor shall replace
it upon return of the defective product,
postage paid, within ninety days of the
date of delivery. There are no other
warranties implied or expressed,
including but not limited to, any implied
warranties of merchantability or fitness
for a particular use.

The warranty registration card must
be on file for repair or replacement
under warranty. It is assumed that any
SNAPSHOT64 package that is returned
without proof of purchase is void of
warranty. This warranty shall be void
if, in the opinion of the authors, this
product has been misused, improperly
installed, modified or otherwise tampered
with.

Your SNAPSHOT64 package should be
sent to:

CSM SOFTSWARE INC.
P.O. Box 563
CROWN POINT, IN. 46307
219-663-4335

Please make sure that your SNAPSHOT64
package is protectively packaged as
damage due to shipping is not covered by
warranty. We would suggest that you
insure your SNAPSHOT64 package.

DISCLAIMER

This product is meant expressly for
the archival backup of your legitimate
software.

Neither the authors nor the
distributors of SNAPSHOT64 condone the
use of this product to assist in software
piracy.

IMPORTANT NOTICE

Use of this product may not be legal in
the state of Louisianna.

HOW TO GET THE MOST OUT YOUR SNAPSHOT 64

This section gives some personal observations on how to get the most from your SNAPSHOT 64. We know many users will discover other techniques too. First and foremost, pay attention to how the program loads in. If you can, watch the head move from track to track. Watch the screen and the border changes. Listen to the speaker on your TV or monitor. Be sure to make written notes about where and when things happen, especially if you're having trouble making the backup.

1. Watch the head move. Go ahead and take the cover off of your drive so that you can see the position of the head. If you know where the protection key is located on the disk, you will be able to make an educated guess as to when the program makes its protection check. If you know that the program checks its protection early on, use your SNAPSHOT 64 to stop the program as soon as the program pauses or when the disk drive stops spinning. If the program loads in a chunk of the main program, then checks the protecion and loads in another chunk of the main program, try to use your SNAPSHOT 64 just before the second chunk of the program gets loaded in. This point can usually be found by watching the movement of the head.

2. Watch for changes in the screen or border of the monitor, while the program is loading. Many times the programmer will set up the program so the screen changes right after the protection check. Occasionally it will be no more than a flicker or a slight change of color. Other times it will be more dramatic, such as a title screen or menu. Remember, programmers are human too - they like to know if a protection scheme is working properly. By watching the screen, many times you will find where the programmer gives himself away.

3. Listen for the sound chip to be initialized. This will normally be accompanied by a small 'POP' from the speaker. If your program is dependent upon sound (or even if it's not) for its proper functioning you may find that by stopping the program just before the sound chip is initialized you will be in the proper spot. Many times you will have to turn the volume way up on your monitor to hear the 'POP'.

4. **Try doing a soft RESET (SYS 64738) after clearing memory with the F3 key. Some programs actually check for the exact start-up screen, and they'll notice the "CLEARED" message.**

If your program works properly, but no sound, try getting the program to go back to the menu. Many programs are designed to restart the menu by using the RUN/STOP and/or the RESTORE keys. Use your owners manual as a guide when finding the proper sequence to restart the program. Occasionally there may be only a second or two where program may be stopped. Try the old "trial and error" method of stopping the program. Make notes of how it functions and what results are obtained. Usually you can narrow down the proper point.

Use the CODE INSPECTOR function of SNAPSHOT 64 to find where the program is executing at. Many times you will stop the program in the middle of a KERNAL routine. Then you will need to examine the STACK to find out what part of the program called the KERNAL routine. (The stack pointer actually points one byte lower on the stack than the last bytes used. The return addresses on the stack must be incrememented by one to be correct.) Look in the general area of the program's execution location for the protection scheme.